AN INVESTIGATION OF THE COMPUTATION OF UPPER CONFIDENCE LEVELS IN A SERIES SYSTEM

Brent Dean Foster

NAVAL POSTGRADUATE SCHOOL
Monterey, California

THESIS

AN INVESTIGATION OF THE COMPUTATION OF UPPER CONFIDENCE LEVELS IN A SERIES SYSTEM

by

Brent Dean Foster

Thesis Advisor: W. M. Woods

March 1973

An Investigation of the Computation of Upper Confidence Levels in a Series System

by

Brent Dean Foster
Lieutenant Commander, United States Navy
B.S., University of Wyoming, 1961

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN OPERATIONS RESEARCH

from the

NAVAL POSTGRADUATE SCHOOL
March 1973

ABSTRACT

A comparison of several techniques is presented for determining upper confidence levels for a system failure rate. A series system of components with exponential failure rates is examined. Classical computational techniques are compared with Bayesian techniques in determining the upper confidence level of a system failure rate. A sensitivity analysis is conducted on several of the parameters as part of the comparison.

TABLE OF CONTENTS

LIST OF TABLES

TABLE OF ABBREVIATIONS

α_i	Optimistic shape parameter of the i^{th} component
β_i	Optimistic scale parameter of the i^{th} component
t_i	Total test time of the i^{th} component
f_i	Number of times the i^{th} component fails
λ_i	Estimate of the failure rate of the i^{th} component
λ_s	Estimate of the system failure rate
λ_u	Upper confidence level on the system failure rate
γ	Level of significance
k	Number of components in the series system
R_s	System reliability
R_l	Lower confidence limit on the system reliability

I. INTRODUCTION

A. BACKGROUND

Numerous "classic" techniques have been used to compute estimates of failure rate and mean time of failure. From these estimates standard accepted procedures can be applied to establish upper confidence levels (UCL) for the failure rate or lower confidence levels (LCL) for the reliability.

One well established "classic" procedure is to utilize the computational methods set forth in Ref. 3 to determine upper confidence levels on system failure rates. This is the procedure used for all "Classic" and "Semi-Classic" methods presented in this paper.

The application of a "Bayesian" technique may be intuitively appealing to some. Results from previous experiments and testing could be applied apriori to current testing programs to determine failure rate and reliability. By using a prior in such a manner perhaps total system test time, number of component failures, etc., could be reduced, thereby reducing the overall expense involved with system life testing. This would appear to be most appealing when testing systems that are extremely expensive.

The intent of this thesis is to attempt to determine if it would be more advantageous to use some "Bayesian" technique rather than a more traditional "classic" technique to compute upper confidence levels on a system failure rate

when the "prior" for the Bayesian method could be chosen as "optimistically" as one would desire.

B. SYSTEM ASSUMPTIONS

Since it is the technique of computation of the upper confidence level that is being investigated, the system that is being modeled can be kept simple, yet still realistic.

Even the most complicated system can be broken down to a system of components, connected in series, which the total failure of any one component will result in the mission failure of the system.

Each type component in this series system experiences exponential failure rate. The failure of each type of component is assumed to be independent of the failure of any other type of component. Wearin and wearout are neglected and the failure rate is assumed to be constant. Components are assumed not to ever be "stillborn." To keep the system as simple as possible, each type component exhibits an identical failure rate, $\lambda_1 = \lambda_i$; $i = 2 \ldots k$, k equaling the number of type components that are connected in the series system. The following formulas concerning system failure rate and system reliability are assumed to be valid for this system:

$$\lambda_s = \sum_{i=1}^{k} \lambda_i$$

$$R_s = R_1 \quad \text{or } R_s = \exp^{-\lambda_s}$$

In keeping the system as simple as possible the optimistic "prior" for the Bayesian computations are also chosen to be equal for each type component, $\alpha_1 = \alpha_i$ and $\beta_1 = \beta_i$, for i = 1, 2, . . . k.

For purposes of comparison, it is assumed that each type of component is placed on test for an equal length of time, $t_1 = t_i$, and each type of component experiences the same number of failures during its total test time, $f_1 = f_i$. This assumption is modified slightly to be able to examine the system that has only one component failure. A system exhibiting two component failures is also examined.

The data (values of the parameters) have been chosen by the author. They have been intentionally chosen to be computationally simple yet to still exhibit the characteristics of a realistic system.

II. COMPUTATIONAL METHODS

A. CLASSIC

Reference 3 is used exclusively for the computations in this method. The computational formulas given in this reference are modified slightly to accommodate the basic assumptions of identical test time, identical number of failures for each component, etc.

$\hat{\lambda}_i$ is a Maximum Likelihood Estimate for the failure rate of the i^{th} component. It will be equal to the total number of failures divided by the total test time.

$$\hat{\lambda}_i = \frac{f_i}{t_i} .$$

$\hat{\lambda}_s$, the system failure rate, will be equal to the sum of the individual component failure rates. $\hat{\lambda}_s = \sum_{i=1}^{k} \hat{\lambda}_i$.

Each component is assumed to fail independently and k will equal the number of components that are placed together in a series.

$$\lambda_u = \frac{2\hat{\lambda}_s + K'^2C + (4\hat{\lambda}_s K'^2 C + K'^4 C^2)^{1/2}}{2}$$

λ_u will be an upper confidence level on the system failure rate.

$$C = \frac{\sum_{i=1}^{k} \frac{\hat{\lambda}_i}{t_i}}{\hat{\lambda}_s}$$

The values for K' are found in Table I. The computational procedure to determine the Beta values for a 90 per cent level of significance are also discussed. Eighty per cent level of significance values are found in Ref. 3.

A formula for the upper confidence level for the system failure rate is also provided when no failures have occurred during the total test time that each component has been allotted. That formula follows:

$$\lambda_u = \frac{K'^2}{n} \sum_{i=1}^{k} \left(\frac{1}{t_i}\right)$$

where n equals the number of component terms in the summation.

Substitution of the upper limits on the failure rates obtained will generate corresponding lower confidence limits on reliability; thus

$$R_1 = \exp^{-\lambda_u} .$$

Beta can be calculated from the following formula:

$$X_u - X = \mathrm{Beta}(K)(X_u)^{1/2}$$

X_u is obtained from Chi-Squared tables in Ref. 1 at the desired confidence level with 2(X+1) degrees of freedom where X is the number of failures and K is the percentage point of the normal distribution point in the same confidence level.

TABLE I

BETA VALUES FOR 90 PER CENT CONFIDENCE

X No. of Failures	$X^2_{.90,2(X+1)}$	$(1/2X^2_{.90,2(X+1)})$ X_μ	Beta Value	K' (Beta(1.282))
0	4.906	2.3025	1.18362	1.5174
1	7.779	3.889	1.14271	1.46496
2	10.645	5.3225	1.12336	1.44015
3	13.362	6.681	1.1109	1.42417
4	15.987	7.994	1.10188	1.41261
5	18.549	9.275	1.09494	1.403713
10	30.813	15.406	1.0743	1.37725
20	54.090	27.045	1.05669	1.35468
30	I N T E R P O L A T E D			1.34639
36.5	91.1	45.55	1.04596	1.34092

The "Beta Value" listed in this table is germane to Ref. 3 and should not be confused with the β that is the optimistic scale parameter used in the Bayesian simulation.

B. BAYESIAN

The Bayesian technique will assume the apriori density to be Gamma$(\lambda_i;\alpha_i,\beta_i)$. α_i and β_i are the shape and scale parameters. The "prior" can be made "optimistic" if the scale parameter is chosen large in relation to the shape parameter.

The a posteriori density then becomes $(\lambda_i;\alpha_i+f_i,\beta_i+t_i)$. f_i will equal the number of failures of the i^{th} type component and t_i will equal the total test time for the i^{th} type component.

The distribution of $\lambda_s = \sum_{i=1}^{k} \lambda_i$ will then be determined by computer simulation. Random variates of each λ_i are generated from the Gamma distribution with parameters $\alpha_i + f_i$, $\beta_i + t_i$. A random variate will be generated for each λ_i, $i = 1, \ldots k$, the number of components in series. The λ_i's will then be added to determine the series system failure rate.

The process for generating a system failure rate is then repeated 1000 times to yield $\lambda_{s1}, \lambda_{s2}, \ldots \lambda_{s1000}$. The 1000 random values of λ_s are then ordered to yield $\lambda_{s(1)}, \lambda_{s(2)}, \ldots \lambda_{s(1000)}$.

$\lambda_{su(\gamma)}$ is the "estimate" of the $(1 - \gamma)^{th}$ percentile point of the distribution of λ_s.

The Bayesian $100(1 - \gamma)^{th}$ upper confidence level for λ_s is then $\lambda_{s1000(1 - \gamma)}$.

At the time of this writing a subroutine to generate random variates from the Gamma distribution does not exist in the Naval Postgraduate School computer library. Reference 2 was used to generate random variates from the Gamma distribution with an integer shape parameter.

The generation of random variates with a shape parameter that is not an integer poses a much more complicated problem. Reference 4 has been written to handle this situation for shape parameters between 0.05 and 1.0. Although LT Robinson's approach is relatively untried, it shows a great deal of promise and may possibly be incorporated into the Naval Postgraduate School computer library in the future. It has been used in this writing for comparative purposes for a shape parameter less than 1.0.

C. SEMI-CLASSIC

Similar to the Classic technique, Ref. 3 is used to compute the upper confidence levels for the system failure rate for this method. The computations will be modified somewhat by adding the identical "optimistic" priors used in the Bayesian technique.

The number of failures per component will be added to α_i and the total test time per component will be added to β_i. A maximum Likelihood Estimate for the i^{th} component's failure is then:

$$\hat{\lambda}_i = \frac{f_i + \alpha_i}{t_i + \beta_i}$$

The system failure rate for this Semi-Classic method is the same as for the Classic method.

$$\hat{\lambda}_s = \sum_{i=1}^{k} \hat{\lambda}_i$$

The value of C will also change slightly:

$$C = \frac{\dfrac{\hat{\lambda}_i}{t_i + \beta_i}}{\hat{\lambda}_s}$$

When no failures have occurred, the upper confidence limit on the failure rate now is:

$$\lambda_u = \frac{K'^2}{n} \qquad \frac{1}{t_i + \beta_i}$$

The computation for the upper confidence level with failure remains the same.

This method is somewhat "Bayesian" in the sense that it is utilizing a prior but it is "Classic" because of the nature of the computational method.

III. PARAMETERS OF VARIATION

Although the range through which some of the parameters are varied could be more extensive, it does provide the reader with an adequate base for comparing the methods of computation.

α_i	1.0, 0.5
β_i	50, 000
t_i	30, 50, 100 mission units
f_i	1.0, 0.0 The failures are also modified so that $f_1 = 1$, $f_2 = f_i = 0$ and $f_1 = 1$, $f_2 = 1$, $f_3 = f_i$, $f_i^{\cdot} = 0$. This will provide the reader with the opportunity to compare a system where only one component or only two components fail.
k	1, 2, 5, 10, 30 components
γ	0.10

IV. COMPARISONS OF SYSTEM FAILURE RATE UPPER CONFIDENCE LEVELS

The following 12 tables provide the reader with upper confidence levels computed from the same data by the three different methods: Classic, Semi-Classic, and Bayesian.

A program to generate random variates from a Gamma distribution with a shape parameter that was greater than 1.0 but not an integer was not available; therefore UCL's for the Bayesian simulation could not be computed for the case when the shape parameter was less than 1.0 and a failure existed for the i^{th} component.

All calculations were conducted on the IBM 360 computer with the exception of the Classic method with zero failures, whose calculations were computed on a desk calculator.

For each three-line block of numbers the reader may compare the three methods' upper confidence levels computed from the same arguments. To see how a change in the scale parameter, the number of times a component fails, or the number of components in series is reflected in the upper confidence level, the reader merely moves to the right or down in the table. If the reader wants to see how a change in the shape parameter or a change in the amount of time a component is on test is reflected, he must go to another table.

TABLE II

SYSTEM FAILURE RATE UPPER CONFIDENCE LEVELS ALFA$_i$ = 1.0

Time$_i$	Beta$_i$	Fail$_i$	k	1	2	5	10	30
30	50	0	CLASSIC	.076750	.076750	.076750	.076750	.076750
			SEMI CLASSIC	.048790	.066531	.115937	.192571	.478729
			BAYESIAN	.026661	.046493	.097500	.172700	.460052
		1	CLASSIC	.130107	.174417	.309166	.513524	1.276610
			SEMI CLASSIC	.067770	.101229	.194152	.339753	.891556
			BAYESIAN	.046493	.083120	.172700	.321967	.868478
	100	0	CLASSIC	.076750	.076750	.076750	.076750	.076750
			SEMI CLASSIC	.030025	.040942	.071346	.118505	.294603
			BAYESIAN	.016407	.028611	.060000	.106277	.283109
		1	CLASSIC	.130107	.174417	.309166	.513524	1.276610
			SEMI CLASSIC	.041705	.062295	.119478	.209079	.548650
			BAYESIAN	.028611	.051577	.106277	.198133	.534447

TABLE III

SYSTEM FAILURE RATE UPPER CONFIDENCE LEVELS $ALFA_i = 1.0$

$Time_i$	$Beta_i$	$Fail_i$	k	1	2	5	10	30
50	50	0	CLASSIC	.046050	.046050	.046050	.046050	.046050
			SEMI CLASSIC	.039032	.053225	.092750	.154057	.382984
			BAYESIAN	.021329	.037194	.078000	.138160	.368041
		1	CLASSIC	.078064	.106450	.185500	.308114	.765967
			SEMI CLASSIC	.054216	.080983	.155321	.271802	.713245
			BAYESIAN	.037194	.067050	.138160	.257573	.694782
	100	0	CLASSIC	.046050	.046050	.046050	.046050	.046050
			SEMI CLASSIC	.026021	.035483	.061833	.102705	.255322
			BAYESIAN	.014219	.024796	.052000	.092107	.245361
		1	CLASSIC	.078064	.106450	.185500	.308114	.765967
			SEMI CLASSIC	.036144	.053989	.103548	.181202	.475497
			BAYESIAN	.024796	.044700	.092107	.171715	.463188

TABLE IV

SYSTEM FAILURE RATE UPPER CONFIDENCE LEVELS ALFA$_i$ = 1.0

Time$_i$	Beta$_i$	Fail$_i$	k	1	2	5	10	30
100	50	0	CLASSIC	.023025	.023025	.023025	.023025	.023025
			SEMI CLASSIC	.026021	.035483	.061833	.102705	.255322
			BAYESIAN	.014219	.024796	.052000	.092107	.245361
		1	CLASSIC	.039032	.053225	.092750	.154057	.382984
			SEMI CLASSIC	.036144	.053989	.103548	.181202	.475497
			BAYESIAN	.024796	.044700	.092107	.171715	.463188
	100	0	CLASSIC	.023025	.023025	.023025	.023025	.023025
			SEMI CLASSIC	.019516	.026612	.046375	.077029	.191492
			BAYESIAN	.010665	.018597	.039000	.069080	.184021
		1	CLASSIC	.039032	.053225	.092750	.154057	.382984
			SEMI CLASSIC	.027108	.040492	.077661	.135901	.356623
			BAYESIAN	.018597	.033525	.069080	.128787	.347391

TABLE V

SYSTEM FAILURE RATE UPPER CONFIDENCE LEVELS ALFA$_i$ = 0.5

Time$_i$	Beta$_i$	Fail$_i$	k	1	2	5	10	30
30	50	0	CLASSIC	.076750	.076750	.076750	.076750	.076750
			SEMI CLASSIC	.038477	.047646	.073919	.114634	.264620
			BAYESIAN	.018439	.028627	.057292	.101570	.258735
		1	CLASSIC	.130107	.177417	.309166	.513524	1.276610
			SEMI CLASSIC	.058481	.084230	.155671	.267077	.686737
			BAYESIAN	*******	*******	*******	*******	*******
	100	0	CLASSIC	.076750	.076750	.076750	.076750	.076750
			SEMI CLASSIC	.023678	.029321	.045488	.070544	.162843
			BAYESIAN	.011347	.017617	.035257	.062505	.159222
		1	CLASSIC	.130107	.177417	.309166	.513524	1.276610
			SEMI CLASSIC	.035988	.051834	.095797	.164355	.422608
			BAYESIAN	*******	*******	*******	*******	*******

TABLE VI

SYSTEM FAILURE RATE UPPER CONFIDENCE LEVELS $ALFA_i = 0.5$

$Time_i$	$Beta_i$	$Fail_i$	k	1	2	5	10	30
50	50	0	CLASSIC	.046050	.046050	.046050	.046050	.046050
			SEMI CLASSIC	.030078	.038117	.059135	.091708	.211696
			BAYESIAN	.014751	.022902	.045834	.081256	.206988
		1	CLASSIC	.078064	.106450	.185500	.308114	.765967
			SEMI CLASSIC	.046785	.067384	.124537	.213661	.549390
			BAYESIAN	*******	*******	*******	*******	*******
	100	0	CLASSIC	.046050	.046050	.046050	.046050	.046050
			SEMI CLASSIC	.020521	.025411	.039423	.061138	.141131
			BAYESIAN	.009834	.015268	.030556	.054171	.137992
		1	CLASSIC	.078064	.106450	.185500	.308114	.765967
			SEMI CLASSIC	.031190	.044923	.083024	.142441	.366260
			BAYESIAN	**	*******	*******	*******	*******

TABLE VII

SYSTEM FAILURE RATE UPPER CONFIDENCE LEVELS $ALFA_i = 0.5$

$Time_i$	$Beta_i$	$Fail_i$	k	1	2	5	10	30
100	50	0	CLASSIC	.023025	.023025	.023025	.023025	.023025
			SEMI CLASSIC	.020521	.025411	.039423	.061138	.141131
			BAYESIAN	.009834	.015268	.030556	.054171	.137992
		1	CLASSIC	.039032	.053225	.092750	.154057	.382984
			SEMI CLASSIC	.031190	.044923	.083024	.142441	.366260
			BAYESIAN	*******	*******	*******	*******	*******
	100	0	CLASSIC	.023025	.023025	.023025	.023025	.023025
			SEMI CLASSIC	.015391	.019058	.029568	.045854	.105848
			BAYESIAN	.007376	.011451	.022917	.040628	.103494
		1	CLASSIC	.039032	.053225	.092750	.154057	.382984
			SEMI CLASSIC	.023390	.033692	.062268	.106831	.274695
			BAYESIAN	*******	*******	*******	*******	*******

TABLE VIII

MODIFIED UPPER CONFIDENCE LEVELS ON SYSTEM FAILURE RATES $ALFA_i = 1.0$

$Time_i$	$Beta_i$	$Fail_i$	k	1	2	5	10	30
30	50	$f_1 = 1$, $f_2 = f_i$, $f_i = 0$	CLASSIC	.130107	.130107	.130107	.130107	.130107
			SEMI CLASSIC	.067770	.136695	.192806	.279285	.594334
			BAYESIAN	.046493	.065268	.114501	.187769	.472251
		$f_1 = 1$, $f_2 = 1$, $f_3 = f_i$, $f_i = 0$	CLASSIC	*******	.177417	.177417	.177417	.177417
			SEMI CLASSIC	*******	.152770	.207180	.292108	.604415
			BAYESIAN	*******	.083812	.129434	.202936	.489156
	100	$f_1 = 1$, $f_2 = f_i$, $f_i = 0$	CLASSIC	.130107	.130107	.130107	.130107	.130107
			SEMI CLASSIC	.041705	.113440	.150096	.206547	.410319
			BAYESIAN	.028611	.040165	.070462	.115550	.290616
		$f_1 = 1$, $f_2 = 1$, $f_3 = f_i$, $f_i = 0$	CLASSIC	*******	.177417	.177417	.177417	.177417
			SEMI CLASSIC	*******	.122974	.158538	.213918	.415675
			BAYESIAN	*******	.051577	.079652	.124883	.301019

TABLE IX

MODIFIED UPPER CONFIDENCE LEVELS ON SYSTEM FAILURE RATES ALFA$_i$ = 1.0

Time$_i$	Beta$_i$	Fail$_i$	k	1	2	5	10	30
50	50	$f_1 = 1$, $f_2 = f_i$, $f_i = 0$	CLASSIC	.078064	.078064	.078064	.078064	.078064
			SEMI CLASSIC	.054216	.093570	.136890	.203823	.449300
			BAYESIAN	.037194	.052215	.091601	.150216	.377801
		$f_1 = 1$, $f_2 = 1$, $f_3 = f_i$, $f_i = 0$	CLASSIC	*******	.106450	.106450	.106450	.106450
			SEMI CLASSIC	*******	.106450	.148479	.214278	.457804
			BAYESIAN	*******	.067050	.103548	.162349	.391324
	100	$f_1 = 1$, $f_2 = f_i$, $f_i = 0$	CLASSIC	.078064	.078064	.078064	.078064	.078064
			SEMI CLASSIC	.036144	.078064	.108433	.155206	.325173
			BAYESIAN	.024796	.034810	.061067	.100144	.251867
		$f_1 = 1$, $f_2 = 1$, $f_3 = f_i$, $f_i = 0$	CLASSIC	*******	.106450	.106450	.106450	.106450
			SEMI CLASSIC	*******	.086603	.116048	.161966	.330403
			BAYESIAN	*******	.044700	.069032	.108232	.260883

TABLE X

MODIFIED UPPER CONFIDENCE LEVELS ON SYSTEM FAILURE RATES ALFA$_i$ = 1.0

Time$_i$	Beta$_i$	Fail$_i$	k	1	2	5	10	30
100	50	$f_1 = 1$, $f_2 = f_i$, $f_i = 0$	CLASSIC	.039032	.039032	.039032	.039032	.039032
			SEMI CLASSIC	.036144	.054216	.082107	.125363	.285135
			BAYESIAN	.024796	.034810	.061067	.100144	.251867
		$f_1 = 1$, $f_2 = 1$, $f_3 = f_i$, $f_i = 0$	CLASSIC	*******	.053225	.053225	.053225	.053225
			SEMI CLASSIC	*******	.062739	.089830	.132403	.291025
			BAYESIAN	*******	.044700	.069032	.108232	.260883
	100	$f_1 = 1$, $f_2 = f_i$, $f_i = 0$	CLASSIC	.039032	.039032	.039032	.039032	.039032
			SEMI CLASSIC	.027108	.046785	.068445	.101911	.224650
			BAYESIAN	.018597	.026107	.045800	.075108	.188900
		$f_1 = 1$, $f_2 = 1$, $f_3 = f_i$, $f_i = 0$	CLASSIC	*******	.053225	.053225	.053225	.053225
			SEMI CLASSIC	*******	.053225	.074240	.107139	.228902
			BAYESIAN	*******	.033535	.051774	.081174	.195662

TABLE XI

MODIFIED UPPER CONFIDENCE LEVELS ON SYSTEM FAILURE RATES $ALFA_i = 0.5$

$Time_i$	$Beta_i$	$Fail_i$	k	1	2	5	10	30
30	50	$f_1 = 1$, $f_2 = f_i$, $f_i = 0$	CLASSIC	.130107	.130107	.130107	.130107	.130107
			SEMI CLASSIC	.054810	.116621	.146407	.192806	.361256
			BAYESIAN	*******	*******	*******	*******	*******
		$f_1 = 1$, $f_2 = 1$, $f_3 = f_i$, $f_i = 0$	CLASSIC	*******	.177417	.177417	.177417	.177417
			SEMI CLASSIC	*******	.133609	.162117	.207180	.373106
			BAYESIAN	*******	*******	*******	*******	*******
	100	$f_1 = 1$, $f_2 = f_i$, $f_i = 0$	CLASSIC	.130107	.130107	.130107	.130107	.130107
			SEMI CLASSIC	.035988	.100392	.119775	.150096	.259836
			BAYESIAN	*******	*******	*******	*******	*******
		$f_1 = 1$, $f_2 = 1$, $f_3 = f_i$, $f_i = 0$	CLASSIC	*******	.177417	.177417	.177417	.177417
			SEMI CLASSIC	*******	.110467	.129082	.158538	.266507
			BAYESIAN	*******	*******	*******	*******	*******

TABLE XII

MODIFIED UPPER CONFIDENCE LEVELS ON SYSTEM FAILURE RATES $ALFA_i = 0.5$

$Time_i$	$Beta_i$	$Fail_i$	k	1	2	5	10	30
50	50	$f_1 = 1$, $f_2 = f_i$, $f_i = 0$	CLASSIC	.078064	.078064	.078064	.078064	.078064
			SEMI CLASSIC	.046785	.078064	.101068	.136890	.267480
			BAYESIAN	*******	*******	*******	*******	*******
		$f_1 = 1$, $f_2 = 1$, $f_3 = f_i$, $f_i = 0$	CLASSIC	*******	.106450	.106450	.106450	.106450
			SEMI CLASSIC	*******	.091662	.113665	.148479	.277238
			BAYESIAN	*******	*******	*******	*******	*******
	100	$f_1 = 1$, $f_2 = f_i$, $f_i = 0$	CLASSIC	.078064	.078064	.078064	.078064	.078064
			SEMI CLASSIC	.031190	.067210	.083321	.108433	.199486
			BAYESIAN	*******	*******	*******	*******	*******
		$f_1 = 1$, $f_2 = 1$, $f_3 = f_i$, $f_i = 0$	CLASSIC	*******	.106450	.106450	.106450	.106450
			SEMI CLASSIC	*******	.076234	.091662	.116048	.205707
			BAYESIAN	*******	*******	*******	*******	*******

TABLE XIII

MODIFIED UPPER CONFIDENCE LEVELS ON SYSTEM FAILURE RATES ALFA = 0.5

$Time_i$	$Beta_i$	$Fail_i$	k	1	2	5	10	30
100	50	$f_1 = 1$, $f_2 = f_i$, $f_i = 0$	CLASSIC	.039032	.039032	.039032	.039032	.039032
			SEMI CLASSIC	.031190	.044243	.059039	.082107	.166657
			BAYESIAN	*******	*******	*******	*******	*******
		$f_1 = 1$, $f_2 = 1$, $f_3 = f_i$, $f_i = 0$	CLASSIC	*******	.053225	.053225	.053225	.053225
			SEMI CLASSIC	*******	.053225	.067384	.089830	.173283
			BAYESIAN	*******	*******	*******	*******	*******
	100	$f_1 = 1$, $f_2 = f_i$, $f_i = 0$	CLASSIC	.039032	.039032	.039032	.039032	.039032
			SEMI CLASSIC	.023392	.039032	.050534	.068445	.133740
			BAYESIAN	*******	*******	*******	*******	*******
		$f_1 = 1$, $f_2 = 1$, $f_3 = f_i$, $f_i = 0$	CLASSIC	*******	.053225	.053225	.053225	.053225
			SEMI CLASSIC	*******	.045831	.056833	.074240	.138619
			BAYESIAN	******	*******	******	******	******

V. CONCLUSIONS

A "crossover" point in this discussion will be defined as the point or general area at which an upper confidence level that was previously lower than one with which it was being compared becomes higher.

The Bayesian method will be compared with the Classic and the Semi-Classic will be compared with Classic. The Bayesian and the Semi-Classic appear to behave similarly and remarks concerning this similarity will be mentioned.

A. UNMODIFIED COMPARISONS, ALFA = 1.0

The first three tables deal with a shape parameter of 1.0 and a system of components that either all of the components experience a failure or none of the components experience a failure.

In the cases where none of the components fail, a crossover point is exhibited in every case when five or more components are in series, regardless of how optimistic the scale parameter becomes or how long the test time is extended. When every component fails, the Bayesian and Semi-Classic systems do not crossover. In both cases the values of the upper confidence level for the Bayesian and the Semi-Classic methods become closer together as the number of components increases.

B. UNMODIFIED COMPUTATIONS, ALFA = 0.5

The upper confidence levels for the second three tables (less the Classic) are computed with a more optimistic prior shape parameter (alfa = 0.5) so naturally the UCL's for these two methods are lower than those shown in the first three tables.

Crossover points are much the same as in the case when alfa = 1.0. The Bayesian and Semi-Classic methods crossover when none of the components fail; only the crossover point occurs when about 10 components are placed in series. The Semi-Classic does not crossover when all components fail (the Bayesian is not examined due to the non-existance of a Gamma random variate generator with a non-integer shape parameter).

C. MODIFIED COMPUTATIONS, ALFA = 1.0

The third three tables provide a much more interesting comparison. Here the situation where only one (or two) component(s) in the system fail. The Semi-Classic and the Bayesian techniques will crossover the Classic system in every case regardless of how optimistic the "priors" are or how long the components are tested. When the priors are more optimistic and/or the test time is larger, then it takes a larger number of components in series for the crossover to occur. This point may occur with as few as two components or as many as between 10 and

30 components. The Bayesian UCL's are strictly lower than the Semi-Classic UCL's.

D. MODIFIED COMPUTATIONS, ALFA = 0.5

In the last three tables only the Semi-Classic and the Classic methods can be compared for the Bayesian cannot be simulated for this case (non-integer).

The Semi-Classic technique is using a more optimistic prior so its UCL's are lower than before but again in every case it will crossover with the Classic method.

Generally one could conclude that as long as the number of components that are in series in a system are few, the Bayesian approach may yield a lower value of an upper confidence level for the system failure rate, provided the priors can be chosen optimistically. If the system is complex enough that more than "a few" components must be placed in series then the Classic system appears to yield the lowest values of upper confidence levels. The Semi-Classic technique would only be useful if the priors were optimistic, the number of components in series were few, and the optimistic priors were non-integers.

COMPUTER PROGRAM FOR THE CLASSIC METHOD

```
C       THE VARIABLE DEFINITIONS FOR THIS PROGRAM FOLLOW:
C          TIME       THE NUMBER OF MISSION UNITS EACH COMPON-
C                     ENT IS UNDER TEST
C          FAIL       THE NUMBER OF TIMES THAT EACH COMPONENT
C                     WILL FAIL DURING THE TOTAL TEST TIME
C          XK         THE NUMBER OF COMPONENTS THAT ARE IN THE
C                     SERIES SYSTEM
C          XLAMI      INDIVIDUAL COMPONENT FAILURE RATE
C          XLAMS      SYSTEM FAILURE RATE
C          XKPRI      A VALUE TAKEN FROM TABLE I (BETA VALUES)
C
C
        XK=1.0
        XKPRI=1.469496
7       TIME=30.0
10      C=0.0
        FAIL=1.0
        XLAMUP=0.0
        XLAMI=FAIL/TIME
        XLAMS=XK*XLAMI
        C=(XK*(XLAMI/TIME))/XLAMS
        XLAMUP=(2.0*XLAMS+(XKPRI**2)*C+SQRT(((4.0*XLAMS)*(XKPR
       1I**2)*C)+(XKPRI**4)*(C**2)))/2.0
        WRITE(6,101)FAIL,XK,TIME,XLAMUP
101     FORMAT(' ',5X,'FAIL=',F7.3,5X,'XK=',F7.3,5X,'TIME=',F7
       2.3,5X,'XLAMUP = ',F10.6,/)
        TIME=TIME+20.0
102     IF(TIME.LE.51.0) GO TO 10
        TIME=TIME+30.0
103     IF(TIME.LE.101.0) GO TO 10
        XK=XK+1.0
        XKPRI=1.44015
104     IF(XK.LE.2.3) GO TO 7
        XK=XK+2.0
        XKPRI=1.403713
105     IF(XK.LT.5.5) GO TO 7
        XK=XK+2.0
        XKPRI=1.37725
106     IF(XK.LT.10.5) GO TO 7
        XK=XK+15.0
        XKPRI=1.34092
107     IF(XK.LT.30.5) GO TO 7
        STOP
        END
```

COMPUTER PROGRAM FOR THE SEMI-CLASSIC METHOD

```
C       THIS SYSTEM IS LABELED "SEMI-CLASSIC" BECAUSE IT
C       UTILIZES THE "PRIORS" THAT WERE INPUTS TO THE BAYES-
C       IAN TECHNIQUE BUT THE COMPUTATIONS ARE PERFORMED THE
C       THE SAME WAY AS IN THE "CLASSIC" TECHNIQUE.
C
C       THE VARIABLE DEFINITIONS FOR THIS PROGRAM FOLLOW:
C           XK          NUMBER OF COMPONENTS IN SERIES
C           FAIL        NUMBER OF FAILURES PER COMPONENT
C           TIME        NUMBER OF MISSION UNITS EACH COMPONENT
C                       IS UNDER TEST
C           BETA        OPTIMISTIC SCALE PARAMETER
C           ALFA        OPTIMISTIC SHAPE PARAMETER
C           XLAMS       SYSTEM FAILURE RATE
C           XLAMI       ESTIMATE OF THE COMPONENT FAILURE RATE
C           XLAMUP      UPPER CONFIDENCE LEVEL ON SYSTEM
C                       FAILURE RATE
C
        ALFA=0.5
6       XK=1.0
        XKPRI=1.469496
7       FAIL=0.0
8       BETA=50.0
9       TIME=30.0
10      C=0.0
        XLAMUP=0.0
        XLAMI=(ALFA+FAIL)/(TIME+BETA)
        XLAMS=XK*XLAMI
        C=(XK*(XLAMI/(TIME+BETA)))/XLAMS
        XLAMUP=((2.0*XLAMS)+((XKPRI**2)*C)+SQRT(((4.0*XLAMS)
       1*(XKPRI**2)*C)+(XKPRI**4)*C**2))/2.0
        WRITE(6,101)ALFA,FAIL,XK,TIME,BETA,XLAMUP
101     FORMAT(' ',2X,'ALFA=',F4.1,2X,'FAIL=',F4.1,2X,'XK='
       6F5.1,2X,'TIME=',F6.1,2X,'BETA=',F6.1,2X,'XLAMUP=',
       7F10.6,//)
        TIME=TIME+20.0
102     IF(TIME.LE.51.0) GO TO 10
        TIME=TIME+30.0
103     IF(TIME.LE.101.0) GO TO 10
        BETA=BETA+50.0
104     IF(BETA.LE.101.0) GO TO 9
        FAIL=FAIL+1.0
        IF(FAIL.LE.1.1) GO TO 8
        XK=XK+1.0
        XKPRI=1.44015
        IF(XK.LE.2.1) GO TO 7
        XK=XK+2.0
        XKPRI=1.403713
        IF(XK.LE.5.1) GO TO 7
        XK=XK+2.0
        XKPRI=1.37725
        IF(XK.LE.10.1) GO TO 7
        XK=XK+15.0
        XKPRI=1.34092
        IF(XK.LE.30.5) GO TO 7
        ALFA=ALFA+0.5
        IF(ALFA.LE.1.1) GO TO 6
        STOP
        END
```

COMPUTER PROGRAM FOR THE MODIFIED SEMI-CLASSIC METHOD

```
C          THIS PROGRAM IS A "MODIFIED SEMI-CLASSIC" IN THAT
C       THE FAILURES ARE MODIFIED SO THAT ONLY ONE COMPONENT
C       OR ONLY TWO COMPONENTS FAIL DURING THE COMPLETE SYS-
C       TEM TEST.  ALL OTHER VARIABLES ARE THE SAME AS FOR
C       THE "SEMI-CLASSIC" PROGRAM.
C
C
C          ONLY ONE SET OF VARIABLES ARE TESTED WITH THIS
C       PROGRAM.  THE VARIABLES TIME,BETA,ALFA, AND FAIL
C       WOULD HAVE TO BE VARIED TO GENERATE A FULL TABLE OF
C       VALUES.
C
C
        XKPRI=1.44015
        XK=2.0
        TIME=100.0
        XLAMS=2.0/100.0
        BETA=0.0
10      ALFA=0.0
        C=1.0/TIME
        XLAMUP=0.0
        XLAMUP=(2.0*XLAMS+(XKPRI**2)*C+SQRT(((4.0*XLAMS)*(XKPR
       1I**2)*C)+(XKPRI**4)*C**2))/2.0
        WRITE(6,101)XK,BETA,TIME,XLAMUP
101     FORMAT(' ',2X,'XK = ',F7.3,2X,'BETA = ',F7.3,2X,
       2'TIME = ',F7.3,2X,'XLAMUP = ',F10.6,/)
        XK=XK+3.0
        XLAMS=2.0/100.0
        IF(XK.LE.5.5) GO TO 10
        XK=XK+2.0
        XLAMS=2.0/100.0
        IF(XK.LE.10.5) GO TO 10
        XK=XK+15.0
        XLAMS=2.0/100.0
        IF(XK.LE.31.0) GO TO 10
        STOP
        END
```

COMPUTER PROGRAM FOR THE BAYESIAN SIMULATION

ALFA = 1.0

```
C       THE DEFINITIONS OF THE VARIABLES USED IN THIS
C       PROGRAM FOLLOW:
C
C          TIME         THE NUMBER OF MISSION UNITS THAT EACH
C                       COMPONENT IS UNDER TEST
C          BETA         THE OPTIMISTIC SCALE PARAMETER
C          ALFA         THE OPTIMISTIC SHAPE PARAMETER
C          FAIL         THE NUMBER OF TIMES THAT EACH COMPONENT
C                       WILL FAIL DURING THE TEST TIME
C          K            THE NUMBER OF COMPONENTS THAT ARE IN THE
C                       SERIES SYSTEM
C          XLAM         THE FAILURE RATE FOR AN INDIVIDUAL
C                       COMPONENT
C          XLAMS        THE SYSTEM FAILURE RATE
C
C
C       THIS SYSTEM IS CALLED
        DIMENSION XLAMS(1000),KEY(1000)
        TIME=30.0
      4 BETA=50.0
      5 FAIL=0.0
      7 IND = 1
    600 DO 610 IO = 1,1000
        XLAMS(IO) = 0.0
    610 CONTINUE
6       ALFA=1.0
        B=TIME+BETA
        KA=ALFA+FAIL
        GO TO (601,602,603,604,606,607),IND
    601 K = 1
        GO TO 608
    602 K = 2
        GO TO 608
    603 K = 5
        GO TO 608
    604 K = 10
        GO TO 608
    606 K = 30
    608 IX=999
9       DO 50 J=1,1000
        DO 609 ILBDS = 1,K
        TR=1.0
10      DO 20 I=1,KA
        CALL RANDU(IX,IY,YFL)
        IX=IY
20      TR=TR*YFL
30      XLAM=-ALOG(TR)/B
        XLAMS(J) = XLAMS(J) + XLAM
    609 CONTINUE
        KEY(J)=J
50      CONTINUE
        CALL SHSORT(XLAMS,KEY,1000)
        WRITE(6,60)K,TIME,BETA,FAIL,XLAMS(900)
60      FORMAT(' ',3X,'COMP. = ',I4,3X,'TIME = ',F7.2,3X,
       1'BETA = ',F7.2,3X,'FAIL = ',F6.2,3X,'LAMS = ',
       2F10.6,//)
        IND = IND + 1
```

```
      GO TO 600
  607 CONTINUE
      FAIL=FAIL+1.0
      IF(FAIL.LE.1.5) GO TO 7
      BETA=BETA+50.0
      IF(BETA.LE.100.1) GO TO 5
      TIME=TIME+20.0
      IF(TIME.LE.50.1) GO TO 4
      TIME=TIME+30.0
      IF(TIME.LE.100.1) GO TO 4
      STOP
      END
```

COMPUTER PROGRAM FOR THE
MODIFIED BAYESIAN SIMULATION
ALFA = 1.0

```
C       THIS BAYESIAN TECHNIQUE IS "MODIFIED" IN MUCH THE
C       SAME WAY THAT THE SEMI-CLASSIC TECHNIQUE WAS IN THAT
C       THE NUMBER OF FAILURES ARE "MODIFIED" SO THAT ONLY
C       ONE AND ONLY TWO COMPONENTS WILL FAIL DURING THE SYS-
C       TEM TEST TIME
C
C
C       THE DEFINITIONS OF THE VARIABLES FOR THIS PROGRAM
C       FOLLOW:
C          TIME         THE NUMBER OF MISSION UNITS THAT EACH
C                       COMPONENT IS UNDER TEST
C          BETA         THE OPTIMISTIC SCALE PARAMETER
C          K            THE NUMBER OF COMPONENTS THAT ARE IN
C                       THE SERIES SYSTEM
C          JJ           THE NUMBER OF COMPONENTS IN THE SYSTEM
C                       THAT FAIL
C          XLAMS        THE SYSTEM FAILURE RATE
C
C
        DIMENSION XLAMS(1000),KEY(1000)
      1 DO 700 JJ=1,2
        TIME=30.0
      4 BETA=50.0
      5 IND=1
C       KA IS EQUAL TO ALFA PLUS FAIL, WHICH IS 2
C       THE INDEX "JJ" IS EQUAL TO THE NUMBER OF COMPONENTS
C       THAT FAIL IN THE SERIES SYSTEM
    600 DO 610 IO=1,1000
        XLAMS(IO)=0.0
    610 CONTINUE
        B=BETA+TIME
        GO TO (601,602,603,604,606,607),IND
    601 K=1
        GO TO 608
    602 K=2
        GO TO 608
    603 K=5
        GO TO 608
    604 K=10
        GO TO 608
    606 K=30
    608 IX=999
        DO 80 J=1,1000
        DO 24 JKJ=1,JJ
        TR=1.0
     10 DO 20 I=1,2
        CALL RANDU(IX,IY,YFL)
        IX=IY
     20 TR=TR*YFL
        XLAMX=-ALOG(TR)/B
        XLAMS(J)=XLAMS(J)+XLAMX
     24 CONTINUE
        IF(K-JJ.LE.0) GO TO 79
        L=K-JJ
        DO 30 ILBDS=1,L
        TR=1.0
C       KA IS NOW EQUAL TO ONE FOR THE REST OF THE SYSTEM
        DO 25 II=1,1
        CALL RANDU(IX,IY,YFL)
```

```
      IX=IY
  25  TR=TR*YFL
      XLAM=-ALOG(TR)/B
      XLAMS(J)=XLAMS(J)+XLAM
  30  CONTINUE
  79  KEY(J)=J
  80  CONTINUE
      CALL SHSORT(XLAMS,KEY,1000)
      WRITE(6,90)K,TIME,BETA,XLAMS(900)
  90  FORMAT(' ',3X,'COMP. = ',I4,3X,'TIME = ',F7.2,3X,
     *'BETA = ',F7.2,3X,'XLAMS = ',F10.6,//)
      IND=IND+1
      GO TO 600
 607  CONTINUE
      BETA=BETA+50.0
      IF(BETA.LE.100.1) GO TO 5
      TIME=TIME+20.0
      IF(TIME.LE.50.1) GO TO 4
      TIME=TIME+30.0
      IF(TIME.LE.100.1) GO TO 4
 700  CONTINUE
      STOP
      END
```

COMPUTER PROGRAM FOR THE BAYESIAN SIMULATION
ALFA = 0.5

```
C       THIS PROGRAM WILL GENERATE VARIATES FROM THE GAMMA
C       DISTRIBUTION WHEN THE SHAPE PARAMETER (ALFA) IS LESS
C       THAN 1.0
C
C
C       THE VARIABLE DEFINITIONS FOR THE MAIN PROGRAM FOLLOW:
C
C          K            THE OPTIMISTIC SHAPE PARAMETER
C          XBETA        THE OPTIMISTIC SCALE PARAMETER
C          XTIME        THE NUMBER OF MISSION UNITS THAT EACH
C                       COMPONENT IS UNDER TEST
C          XFAIL        THE NUMBER OF TIMES THAT EACH COMPONENT
C                       FAILS DURING THE TEST TIME
C          NCMP         THE NUMBER OF COMPONENTS IN THE SERIES
C                       SYSTEM
C          XLAMS        THE SYSTEM FAILURE RATE
C          XLAMUP       THE UPPER CONFIDENCE LEVEL ON THE
C                       FAILURE RATE
C
C
        DIMENSION XLAMS(1000),KEY(1000)
        REAL*4 K
        K=0.5
        XTIME=30.0
   30   XBETA=50.0
   40   IND=1
  600   DO 610 IO=1,1000
        XLAMS(IO)=0.0
  610   CONTINUE
        BETA=1.0/(XBETA+XTIME)
        GO TO(601,602,603,604,605,606),IND
  601   NCMP=1
        GO TO 607
  602   NCMP=2
        GO TO 607
  603   NCMP=5
        GO TO 607
  604   NCMP=10
        GO TO 607
  605   NCMP=30
  607   IX=999
        CALL GMINIT(K,BETA)
   10   DO 50 J=1,1000
        DO 609 ILBDS=1,NCMP
        CALL GAMA(IX,Z)
C       Z WILL = AN UNORDERED VALUE OF LAMBDAI
        XLAMS(J)=XLAMS(J)+Z
  609   CONTINUE
        KEY(J)=J
   50   CONTINUE
C       NOW ORDER THE SYSTEM FAILURE RATES
        CALL SHSORT(XLAMS,KEY,1000)
        WRITE(6,60)NCMP,XBETA,XTIME,XLAMS(900)
   60   FORMAT(' ',2X,'NCMP =',I4,2X,'BETA =',F6.2,2X,
       *'TIME =',F6.2,2X,'XLAMS =',F10.6,/)
        IND=IND+1
        GO TO 600
  606   CONTINUE
        XBETA=XBETA+50.0
```

```
IF(XBETA.LE.100.1) GO TO 40
XTIME=XTIME+20.0
IF(XTIME.LE.50.1) GO TO 30
XTIME=XTIME+30.0
IF(XTIME.LE.100.1) GO TO 30
END
```

```
C
C     SUBROUTINE GMINIT(K,BETA)
C
C     ADDITIONAL ENTRY POINTS:
C
C     RSULT
C     GAMA(IX,Z)
C
C
C     PURPOSE:
C
C     GENERATION OF GAMMA RANDOM DEVIATES WITH SHAPE
C     PARAMETER LESS THAN ONE.
C
C     METHOD:
C
C     A MODIFICATION OF MARSAGLIA'S BOX-WEDGE-TAIL METHOD
C     FOR NORMAL DEVIATES IS USED. THE PDF IS DECOMPOSED
C     INTO A HEAD REGION, A NUMBER (DEPENDENT ON K) OF
C     RECTANGLES AND WEDGES AND A TAIL REGION. THE GMINIT
C     SECTION OF THE SUBROUTINE ALSO SETS UP A BINARY
C     SEARCH TREE TO BE USED FOR EFFICIENT SELECTION OF THE
C     PROPER REGION DURING THE ACTUAL GENERATING PROCESS,
C     WHICH IS HANDLED BY THE GAMA SECTION
C
C     DESCRIPTION OF PARAMETERS
C
C     K       GAMMA DISTRIBUTION SHAPE PARAMETER
C             (MUST BE .GE. 0.05 AND .LE. 1.0)
C     BETA    GAMMA DISTRIBUTION SCALE PARAMETER
C     IX      SEED FOR RANDOM NUMBER GENERATOR
C     Z       RETURNED GAMMA DEVIATE
C
C     THE PDF OF THE GAMMA FUNCTION IS GIVEN BY
C
C     F(X) = (1/BETA)**K * X**(K-1) * EXP(-X/BETA)/GAMMA(K)
C
C     THE FOLLOWING SUBROUTINES ARE USED:
C
C     RANDOM(IX)    RETURNS A UNIFORM (0,1) DEVIATE
C     INVGAM(K,X)   COMPUTES THE INVERSE GAMMA CDF
C     IGAM(K,X)     COMPUTES THE INCOMPLETE GAMMA
C                   FUNCTION(GAMMA CDF)
C
C     NOTE:
C     UNDERFLOW IS POSSIBLE WHEN K IS LESS THAN.18 AND
C        BECOMES MORE LIKELY AS K DECREASES.  WHEN K IS 0.5
C        THE PROBABILITY OF UNDERFLOW IS ABOUT.000129 FOR
C        ANY GENERATED DEVIATE.
C


      SUBROUTINE GMINIT(K,BETA)
      REAL*4 K,INVGAM,IGAM
      INTEGER*4 FIRST,TABLE,BOTTOM,END
      LOGICAL*1 USED
      DIMENSION P(100),X(101),H(100),Q(100),R(100),B(100)
      DIMENSION TABLE(202),PROB(202),NEXT(202),LAST(202)
      DIMENSION TEST(202),LIST(202),USED(202)
      DIMENSION RAND(2)
      EQUIVALENCE (U,RAND(1)),(V,RAND(2))
C
C
C     THIS FIRST SECTION INITIALIZES CONSTANTS AND TABLES TO
C        BE USED BY GAMA WHEN IT IS CALLED.  THE FOLLOWING
C     ARE USED BY GAMA:
C
C     P0      PROBABILITY FOR "HEAD" REGION
C     PN      PROBABILITY FOR "TAIL" REGION
C     P(I)    PROBABILITY FOR I-TH RECTANGLE
C     X(I)    LEFT-HAND BOUNDARY OF I-TH RECTANGLE
C     H(I)    WIDTH OF I-TH RECTANGLE
C     Q(I)    PROBABILITY OF I-TH WEDGE
```

```
C       R(I)     REJECTION TEST RATIO FOR I-TH WEDGE
C       B(I)     Y INTERCEPT FOR I-TH WEDGE
C       ALPHA    SHAPE PARAMETER - 1.
C       PROB     ORDERED VECTOR OF PROBABILITIES
C       TABLE    VECTOR OF WEDGE/RECTANGLE NUMBERS CORRESPOND-
C                ING TO PROBABILITIESS IN PROB
C       FIRST    STARTING POINT FOR BINARY SEARCH
C       NEXT,    LINKS FOR BINARY SEARCH
C        LAST
C       J1       POSITION IN PROB OF P0
C       H1 TO CONSTANTS FOR APPROXIMATION TO INVERSE GAMMA
C        H4      CDF FOR SMALL VALUES OF Z
C
C
C       CHECK FOR K IN RANGE
        IF((K.GE.0.05) .AND. (K.LE.1.0))GO TO 5
        WRITE(6,4)K
      4 FORMAT(//'DGMINIT CALLED WITH K=',1PE16.6,
     *'  OUT OF RANGE'/)
        RETURN
C
C       GET UPPER BOUND ON NUMBER OF RECTANGLES
      5 N=20.+6.6/K
        IF(N.GT.100)N=100
        M=2*N+2
        MM=M-1
        ALPHA=K-1.
        GK=GAMMA(K)
        P0=5.E-5/(K*K)
        HFAC=2.
C
C       SET UP RECTANGLE BOUNDS
C
        X(1)=INVGAM(K,P0)
        P0=IGAM(K,X(1))
        H(1)=.25*X(1)
        DO 10 I=2,N
             X(I)=X(I-1)+H(I-1)
             H(I)=H(I-1)*HFAC
             P(I)=0.
             Q(I)=0.
     10 CONTINUE
        X(N+1)=X(N)+H(N)
C
C       ZERO PROBABILITY VECTORS AND LINKS
        DO 15 I=1,M
             NEXT(I)=0
             LAST(I)=0
             PROB(I)=0.
             LIST(I)=0
             USED(I)=.FALSE.
     15 CONTINUE
C
C       FIND COEFFICIENTS FOR NEWTON-RAPHSON APPROXIMATION TO
C       SLOPE OF PDF
C
        B1=-2.*ALPHA
        B2=ALPHA*(ALPHA-1.)
        A1=B1+1.
        A2=ALPHA*(ALPHA-2.)
        C=1.-ALPHA
C
C       FIND RECTANGLE PROBABILITIES AND WEDGE VALUES
C
        PL=P0
        FL=EXP(ALPHA*ALOG(X(1))-X(1))/GK
        DO 40 I=1,N
             FU=EXP(ALPHA*ALOG(X(I+1))-X(I+1))/GK
             PU=IGAM(K,X(I+1))
             P(I)=H(I)*FU
             Q(I)=PU-PL-P(I)
C
```

```
C
C           NEWTON-RAPHSON ITERATION TO FIND POINT WHERE
C           SLOPE OF PDF IS PARALLEL TO CHORD
C
            W=X(I)
            S=(FU-FL)/H(I)
            SC=S*GK
            DO 20 J=1,8
                 Y=W*((W+A1)*W+A2+SC*EXP(C*ALOG(W)+W))/((W+B1
                 IF(ABS(Y-W).LT.1.E-4*Y)GO TO 30
                 W=Y
   20       CONTINUE
C
C           FIND VALUES FOR REJECTION METHOD
C
   30       B(I)=EXP(ALPHA*ALOG(Y)-Y)/GK+S*(X(I)-Y)
            R(I)=(B(I)-FU)/(FL-FU)
C
C           TEST TO SEE IF ENOUGH RECTANGLES HAVE BEEN TAKEN
            IF(PU.GT.0.999)GO TO 45
C
C           RESET PROBABILITY AND FUNCTION VALUES FOR NEXT
C           RECTANGLE
C
            PL=PU
            FL=FU
   40 CONTINUE
C
C     FIND LOWER END OF NON-ZERO PROBABILITY VECTOR
   45 LOW=2*(N-I)+1
C
C     FIND TAIL PROBABILITY
      SUM1=P0
      DO 50 J=1,N
            SUM1=SUM1+P(J)+Q(J)
   50 CONTINUE
      PN=1.-SUM1
C
C     GET VALUES TO COMPUTE LOW END APPROXIMATION TO INVERSE
C     GAMMA CDF
C
      H1=K*GK*P0
      H2=1./K
      H3=(K+1.)*5.0E-7
      H4=4./(K+1.)
C
C     GENERATE PROBABILITY VECTOR
C
      DO 80 I=1,N
            PROB(I)=P(I)
            TABLE(I)=I
            PROB(I+N)=Q(I)
            TABLE(I+N)=-I
   80 CONTINUE
      PROB(M-1)=P0
      TABLE(M-1)=0
      PROB(M)=PN
      TABLE(M)=0
C
C      SORT PROBABILITY VECTOR
C
      DO 110 I=1,MM
            ICH=0
            L=M-I
            DO 100 J=1,L
                 IF(PROB(J)-PROB(J+1))100,100,90
   90            TEMP=PROB(J)
                 PROB(J)=PROB(J+1)
                 PROB(J+1)=TEMP
                 ITEMP=TABLE(J)
                 TABLE(J)=TABLE(J+1)
                 TABLE(J+1)=ITEMP
                 ICH=1
```

```
  100       CONTINUE
            IF(ICH)120,120,110
  110 CONTINUE
      PROB(M)=1.
C
C     CONVERT PROB TO CUMULATIVE PROBABILITIES AND
C      FIND FIRST AND J1
C
  120 J1=0
      FIRST=0
      L=LOW+1
      DO 130 I=L,M
            IF((TABLE(I).EQ.0) .AND. (PROB(I).EQ.PO))J1=I
            PROB(I)=PROB(I)+PROB(I-1)
            IF((PROB(I).GE.0.5) .AND. (FIRST.EQ.0))FIRST=I
  130 CONTINUE
      IF(FIRST.EQ.M)FIRST=MM
C
      IF(J1)140,140,150
  140 J1=LOW
C
C     NOW DETERMINE THE VECTORS NEXT AND LAST FOR BINARY
C     SEARCH
C
  150 BOTTOM=1
      END=1
      PR=.25
      LIST(1)=FIRST
      TEST(1)=.5
      USED(FIRST)=.TRUE.
  151 DO 159 I=1,BOTTOM
            LI=LIST(I)
            IF(USED(LI+1))GO TO 155
            PRN=TEST(I)+PR
            DO 152 J=LOW,MM
                  IF(PROB(J).GT.PRN)GO TO 153
  152       CONTINUE
  153       IF(.NOT. USED(J))GO TO 154
            J=J-1
            IF(LI-J)153,155,155
  154       NEXT(LI)=J
  155       LIST(I)=NEXT(LI)
            IF((LI.EQ.LOW) .OR. USED(LI-1))GO TO 159
            PRL=TEST(I)-PR
            TEST(I)=PRN
            DO 156 J=LOW,MM
                  IF(PROB(J).GT.PRL)GO TO 157
  156       CONTINUE
  157       IF(.NOT. USED(J))GO TO 1571
            J=J+1
            IF(LI-J)158,158,157
 1571       LAST(LI)=J
            GO TO 1585
  158       J=LI
 1585       END=END+1
            LIST(END)=J
            TEST(END)=PRL
  159 CONTINUE
      BOTTOM=END
      PR=PR*0.5
      I=1
 1591 IF(LIST(I))160,160,163
  160 BOTTOM=BOTTOM-1
      IF(BOTTOM)165,165,161
  161 DO 162 J=I,BOTTOM
            LIST(J)=LIST(J+1)
            TEST(J)=TEST(J+1)
  162 CONTINUE
      GO TO 1591
  163 USED(LIST(I))=.TRUE.
      I=I+1
      IF(I.LE.BOTTOM)GO TO 1591
```

```
      END=BOTTCM
      GO TO 151
C
C     SETUP FIRST CALL TO RANDOM
  165 CALL OVFLOW
C
C     THROUGH WITH SETUP PHASE. QUIT.
      RETURN
C
C
C     THIS SECTION PRINTS OUT THE VALUES GENERATED BY GMINIT
C
      ENTRY RSULT
      WRITE(6,166)K,BETA
  166 FORMAT(/'1GENERATED VALUES FOR K=',1PE14.6,
     *'    BETA=',E14.6)
      WRITE(6,170)P0,PN
  170 FORMAT('OHEAD PROBABILITY=',1PE15.6,
     *'    TAIL PROBABILITY=',E15.6)
      WRITE(6,180)
  180 FORMAT('ORECTANGLE/WEDGE VALUES'//2X,'I',9X,'X(I)',12X
     *,'P(I)',12X,'Q(I)',12X,'R(I)',12X,'B(I)'/)
      SUM1=0.
      SUM2=0.
      DO 192 I=1,N
      IF(P(I))193,193,185
  185 SUM1=SUM1+P(I)
      SUM2=SUM2+Q(I)
      WRITE(6,190)I,X(I),P(I),Q(I),R(I),B(I)
  190 FORMAT(1XI3,1P5E16.6)
  192 CONTINUE
  193 WRITE(6,194)SUM1,SUM2
  194 FORMAT('OSUMS',15X,1P2E16.6)
      WRITE(6,195)J1,H1,H2,H3,H4
  195 FORMAT(/'OVALUES FOR HEAD/TAIL APPROXIMATION:'/' J1=',
     *I6/' H1=',E16.6,'    H2=',E16.6,'    H3=',E16.6,'    H4',
     1E16.6)
      WRITE(6,196)FIRST
  196 FORMAT(/'OSTARTING POINT FOR BINARY SEARCH',I4)
      WRITE(6,197)(I,PROB(I),TABLE(I),NEXT(I),LAST(I),I=1,M)
  197 FORMAT(/'OPROBABILITY VECTOR AND LINKS'//14X,'PROB',9X
     *,'TABLE',4X,'NEXT',5X,'LAST'//(1XI5,1PE16.6,0P3I9))
      RETURN
C
C     THIS IS THE SECTION THAT ACTUALLY COMPUTES THE RANDOM
C     VARIATES
C
      ENTRY GAMA(IX,Z)
C
C     GET TWO UNIFORM DEVIATES
      CALL RANDOM(IX,U,2)
C
C     CONDUCT BINARY SEARCH OF PROBABILITY VECTOR
C
      J=FIRST
  200 IF(U-PROB(J))210,250,230
  210 IF(LAST(J))250,250,220
  220 J=LAST(J)
      GO TO 200
  230 IF(NEXT(J))245,245,240
  240 J=NEXT(J)
      GO TO 200
  245 J=J+1
C
C     LOCATED PROBABILITY DIVISION. GET TABLE VALUE.
  250 N=TABLE(J)
      IF(N)260,290,320
C
C     THIS SECTION IS FOR THE WEDGES
C
  260 N=-N
      CALL RANDOM(IX,U,1)
```

```
  270 IF(U.LE.V)GO TO 280
      TEMP=U
      U=V
      V=TEMP
  280 Z=X(N)+H(N)*U
      IF(V.LE.R(N))GO TO 330
C
C     THIS STEP IS PERFORMED VERY RARELY
      W=U+EXP(ALPHA*ALOG(Z)-Z)/B(N)
      IF(V.LE.W)GO TO 330
      CALL RANDOM(IX,U,2)
      GO TO 270
C
C     THIS SECTION IS FOR HEAD/TAIL PROBABILITIES
C
  290 IF(J.EQ.J1)GO TO 300
      Z=INVGAM(K,1.-PN*V)
      GO TO 330
  300 Z=(H1*V)**H2
      IF(Z.LT.H3)GO TO 330
      Z=2.*Z/(1.+SQRT(1.-H4*Z))
      GO TO 330
C
C     THIS SECTION IS FOR THE RECTANGLES
C
  320 Z=X(N)+H(N)*V
C
C     SCALE GAMMA VARIATE AND EXIT
C
  330 Z=Z*BETA
      RETURN
      END
```

```
C       INVGAM SUBROUTINE

        FUNCTION INVGAM(K,Z)
        REAL*4 INVGAM,IGAM,K,EPS/1.E-08/
        IF(Z.GT.0.0)GO TO 10
        INVGAM=0.0
        RETURN
     10 X=Z
        T=K-1.
        G=GAMMA(K)
        DO 40 I=1,30
        Y=G*(IGAM(K,X)-Z)/(EXP(T*ALOG(X)-X))
     20 IF(Y.LT.X)GO TO 30
        Y=Y*.5
        GO TO 20
     30 X=X-Y
        IF(ABS(Y)-EPS*X)50,50,40
     40 CONTINUE
     50 INVGAM=X
        RETURN
C       THIS IS THE END OF THE INVGAM SUBROUTINE
        END
```

```
C       IGAM SUBROUTINE

        FUNCTION IGAM(K,X)
        IMPLICIT REAL*8 (D)
        REAL*4 IGAM,K
        REAL*8 EPS/1.D-13/
        IF(X.GT.0.)GO TO 5
        IGAM=0.
        RETURN
      5 IF(X.LE.12.0)GO TO 8
        IGAM=1.0
        RETURN
      8 DX=DBLE(X)
        DK=DBLE(K)
        DTERM=DX**DK/DGAMMA(DK)
        DSUM=DTERM/DK
        DO 20 I=1,30
        IF(DABS(DTERM)-EPS*DSUM)40,40,10
     10 DTERM=DTERM*(-DX)/DFLOAT(I)
        DSUM=DSUM+DTERM/(DK+DFLOAT(I))
     20 CONTINUE
     40 IGAM=SNGL(DSUM)
        RETURN
C       THIS IS THE END OF THE IGAM SUBROUTINE
        END
```

BIBLIOGRAPHY

1. Hillier, F. J. and Liberman, G. J., Introduction to Operations Research, p. 626, Holden-Day, Inc., 1967.

2. Naylor, T. H. and others, Computor Simulation Techniques, p. 87-90, John Wiley & Sons, Inc., 1966.

3. Department of the Navy Weapons Specifications OD-29304, Guide Manual for Reliability Measurement Program, 15 May 1965.

4. Robinson, W. E., Trend Analysis in a Renewal Process, M.S. Thesis, Naval Postgraduate School, Monterey, 1973.

INITIAL DISTRIBUTION LIST

		No. Copies
1.	Defense Documentation Center Cameron Station Alexandria, Virginia 22314	2
2.	Library, Code 0212 Naval Postgraduate School Monterey, California 93940	2
3.	Professor W. Max Woods, Code 53 Wo Department of Mathematics Naval Postgraduate School Monterey, California 93940	1
4.	Chief of Naval Personnel Pers 11b Department of the Navy Washington, D. C. 20370	1
5.	Naval Postgraduate School Department of Operations Research and Administrative Sciences Monterey, California 93940	1
6.	Professor Robert R. Read, Code 54 Re Department of Operations Research and Administrative Sciences Naval Postgraduate School Monterey, California 93940	1
7.	LCDR Brent D. Foster, USN 110 Moran Circle Monterey, California 93940	1

UNCLASSIFIED
Security Classification

DOCUMENT CONTROL DATA - R & D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

ORIGINATING ACTIVITY (Corporate author) Naval Postgraduate School Monterey, California 93940	2a. REPORT SECURITY CLASSIFICATION Unclassified 2b. GROUP

REPORT TITLE

An Investigation of the Computation of Upper Confidence Levels in a Series System

DESCRIPTIVE NOTES (Type of report and, inclusive dates)

Master's Thesis; March 1973

AUTHOR(S) (First name, middle initial, last name)

Brent Dean Foster

REPORT DATE	7a. TOTAL NO. OF PAGES	7b. NO. OF REFS
March 1973	52	4

CONTRACT OR GRANT NO.	9a. ORIGINATOR'S REPORT NUMBER(S)
PROJECT NO.	
	9b. OTHER REPORT NO(S) (Any other numbers that may be assigned this report)

DISTRIBUTION STATEMENT

Approved for public release; distribution unlimited.

SUPPLEMENTARY NOTES	12. SPONSORING MILITARY ACTIVITY Naval Postgraduate School Monterey, California 93940

ABSTRACT

A comparison of several techniques is presented for determi ing upper confidence levels for a system failure rate. A scries sy tem of components with exponential failure rates is examined. Clas ical computational techniques are compared with Bayesian techniques in determining the upper confidence level of a system failure rate. A sensitivity analysis is conducted on several of the parameters as part of the comparison.

DD FORM 1 NOV 65 1473 (PAGE 1)

S/N 0101-807-6811

UNCLASSIFIED
Security Classification

A-31408

KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT
Gamma Function						
Bayesian						
Series System						
Reliability						
Upper Confidence Level						
Reliability						
Apriori						
Aposteriori						
Exponential						

Thesis
F653 Foster 143415
c.1
An investigation of the computation of upper confidence levels in a series system.